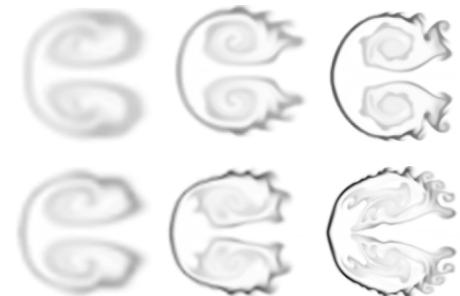
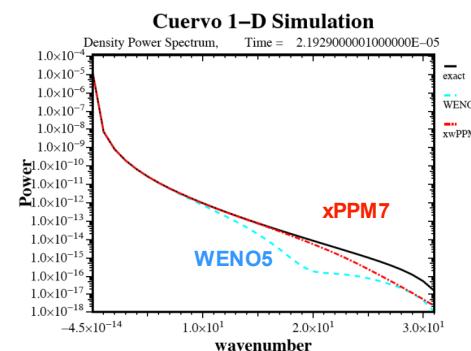
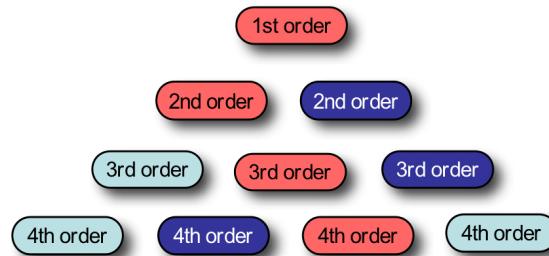


Approaches to Improved Resolution Methods with Eulerian Hydrodynamics



William Rider*, Jeff Greenough[§] and Jim Kamm*

*LANL

§LLNL

5 Lab Meeting, June 19-24, 2005

Vienna, Austria

Question: How do you get the most out your differencing scheme?

Caveats: *our main goal is to compute nonlinear discontinuous flow, shock hydrodynamics, turbulence*

- The goal is get the most accuracy for the least cost—and not sacrifice robustness.
- Its useful to reexamine the design principles used in existing methods – then see how to get the most out of them.



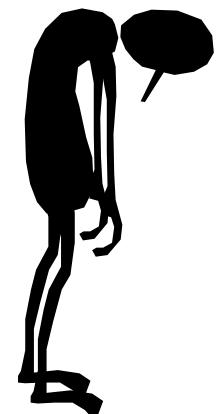
Discontinuities are special: weak solutions have some important requirements*

- *The Lax-Wendroff theorem is one of the few rigorous theoretical results to rely upon,*
 - If the scheme is in conservation form then the solutions converge to a weak solution (weak solutions are not unique!),
 - and with sufficient dissipation, the unique solution can be found.
- **Without conservation, no guarantees!**

*Lax & Wendroff, *Comm. Pure Appl. Math.*, 13, 1960. Also see
R.J. Leveque, *Numerical Methods for Conservation Laws*

Discontinuities are special: first order accuracy is expected.

- For *coupled systems* with *discontinuities* high-order accuracy is lost between characteristics moving away from the discontinuity*
 - Several recent works have re-confirmed this result (Osher, Carpenter, *Greenough & Rider*)
- Generally, with smooth data and a nonlinear system of hyperbolic conservation laws, a discontinuity (i.e., shock) will eventually form
 - *Therefore the loss of accuracy is nearly certain!*



*Majda & Osher, *Comm. Pure Appl. Math.*, 30 1977.

Here is a working definition for the scaling of computational effort

- A function of the cost of a solution on a given grid and the relative accuracy *with same rate of convergence*

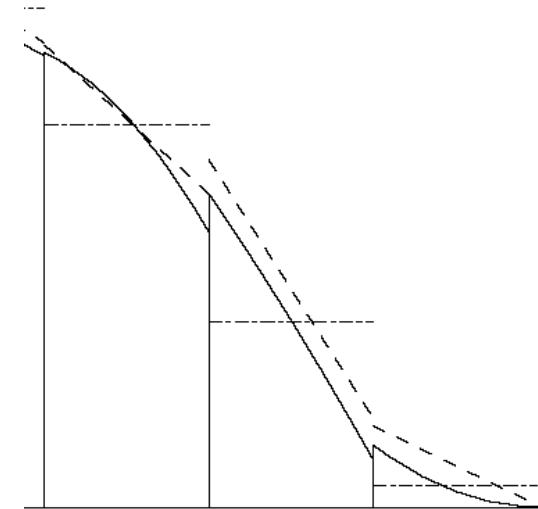
$$\eta = (\text{cost})(\text{RE})^{(d+1)/n}$$

- Where d =dimension, n =convergence rate, RE = Relative error with Error = Ah^n

How much relative effort must be expended to compute a solution of a given accuracy?

I will first discuss the design principles of modern high resolution methods

For the most part, I will talk about designing methods via interpolating variables in a zone.



It's important to remember that methods must balance between accuracy, instability and robustness (dissipation).

The rules associated with high-resolution methods are either rigorous or heuristic

- FCT - a physically based heuristic monotonicity
- MUSCL/PPM - a geometrically based heuristic monotonicity
- TVD - a rigorous algebraic monotonicity
- UNO/ENO/WENO - somewhere between rigor and heuristic

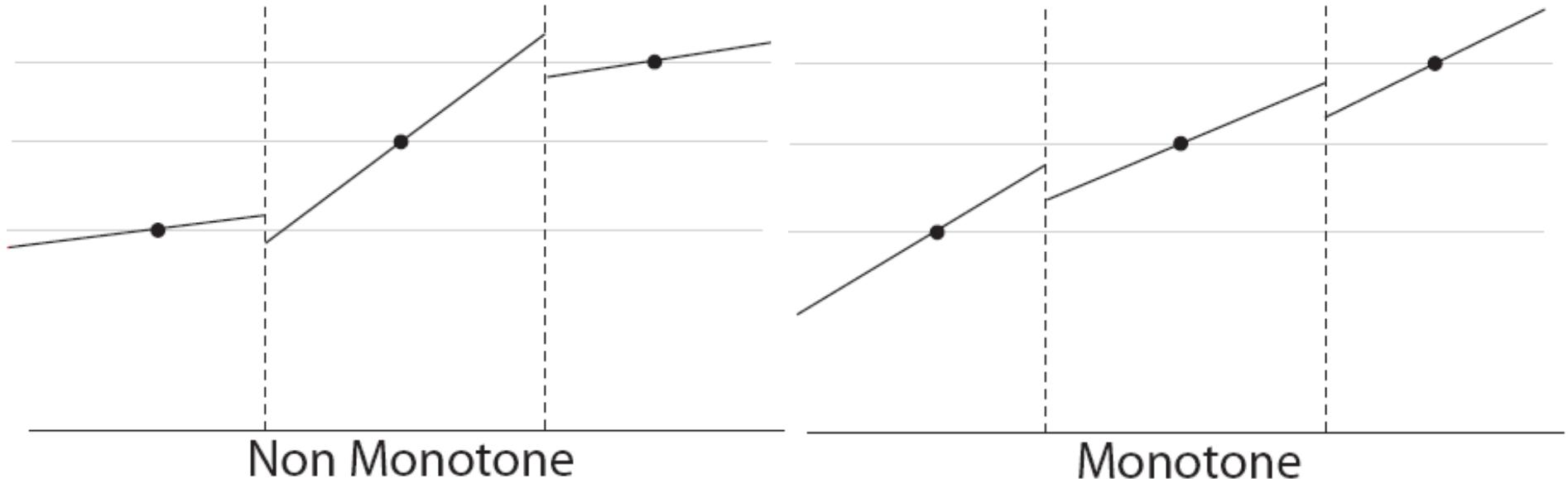
Monotonicity Based

The end result is a nonlinearly stable, high-resolution (not order) method.

What does TVD mean anyway?

- Define TV=total variation $\mathbf{TV} = \sum u_{j+1} - u_j$
- TVD means the *total variation diminishing*
 $\mathbf{TV}(u^{n+1}) \leq \mathbf{TV}(u^n)$ or $\mathbf{TV}(\mathbf{R}(u^n)) \leq \mathbf{TV}(u^n)$
 - Here $\mathbf{R}(u)$ is a reconstruction (interpolation) of u
 - TVD is approximately (algebraically) monotone
- ENO is closely related
 $\mathbf{TV}(\mathbf{R}(u^n)) \leq \mathbf{TV}(u^n) + O(h^{r+1})$
 - Thus ENO is almost monotone with oscillations of a size $O(h^{r+1})$

Monotone methods can be developed using a geometric argument



Non Monotone

Monotone

This can be expressed in an algebraic form

$$u(x) = u_j + s_j \left(\frac{x - x_j}{\Delta x} \right)$$

$$s_j := \text{minmod}(s_j, 2(u_j - u_{j-1}), 2(u_{j+1} - u_j))$$

minmod returns the minimum absolute value of the arguments if they have the same sign.

TVD & FCT are closely related to monotone methods, but algebraic — and limited to 2nd order formal accuracy.

- TVD is an algebraic formula producing, nonlinear upwind methods: $u_t + au_x = 0$

$$u_j^{n+1} = u_j^n - C_{j-1/2} \Delta_{j-1/2} u^n + C_{j+1/2} \Delta_{j+1/2} u^n$$

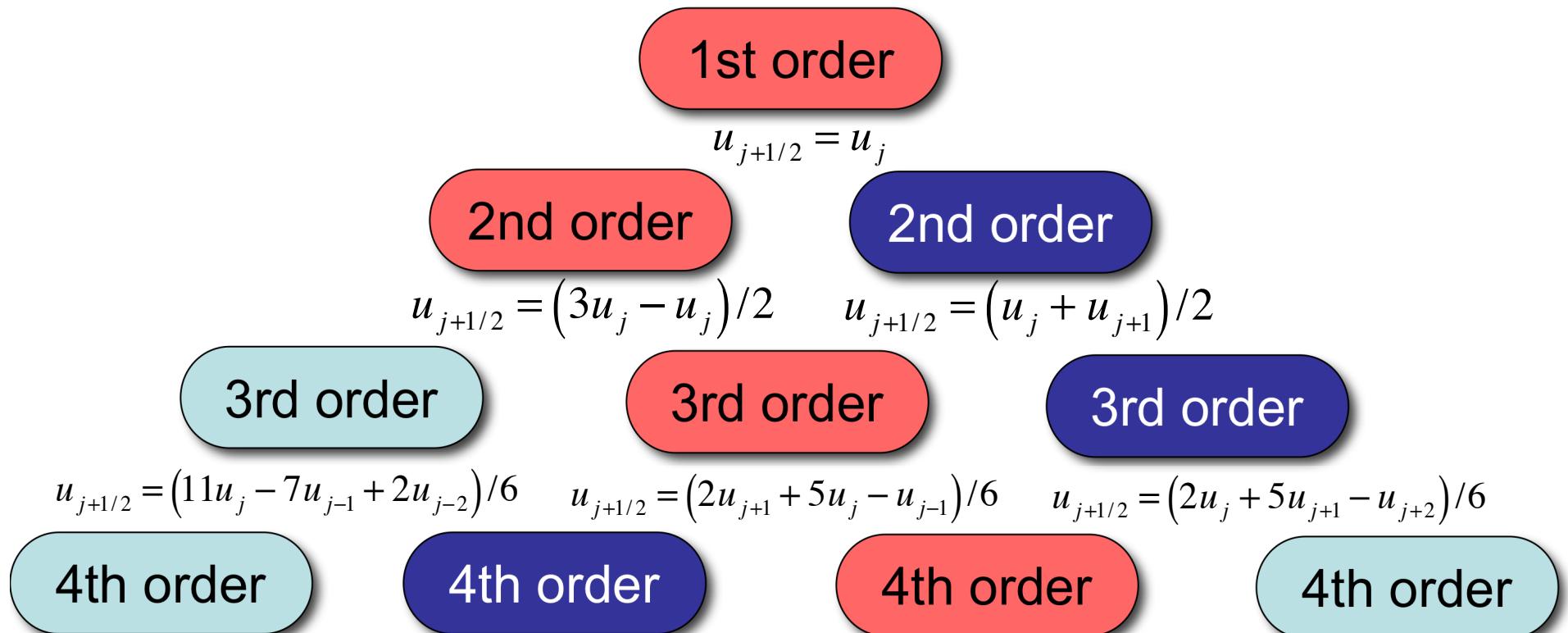
$$C_{j-1/2} = \frac{a\Delta t}{\Delta x}; C_{j+1/2} = 0 \quad C_{j-1/2} \geq 0; C_{j+1/2} \geq 0$$

– Upwind is linear, TVD is nonlinear, $C_{j-1/2}$ is a function of $[.., u_{j-2}, u_{j-1}, u_j, u_{j+1}, ..]$

- The FCT method was the original “limiter” method, combining (i.e. hybridizing) high-order with first-order monotone methods.

ENo Methods use an adaptive stencil that chooses the “smoothest” stencil locally, The method is formally higher order.

LA-UR-05-3733



- ENO selects stencils *adaptively* by choosing the one that is closest to the next lower order.

The same differencing may be arrived at through a different path.

1st order

$$u_{j+1/2} = u_j$$

2nd order

$$u_{j+1/2} = (3u_j - u_j)/2$$

2nd order

$$u_{j+1/2} = (u_j + u_{j+1})/2$$

3rd order

$$u_{j+1/2} = (11u_j - 7u_{j-1} + 2u_{j-2})/6$$

3rd order

$$u_{j+1/2} = (2u_{j+1} + 5u_j - u_{j-1})/6$$

3rd order

$$u_{j+1/2} = (2u_j + 5u_{j+1} - u_{j+2})/6$$

4th order

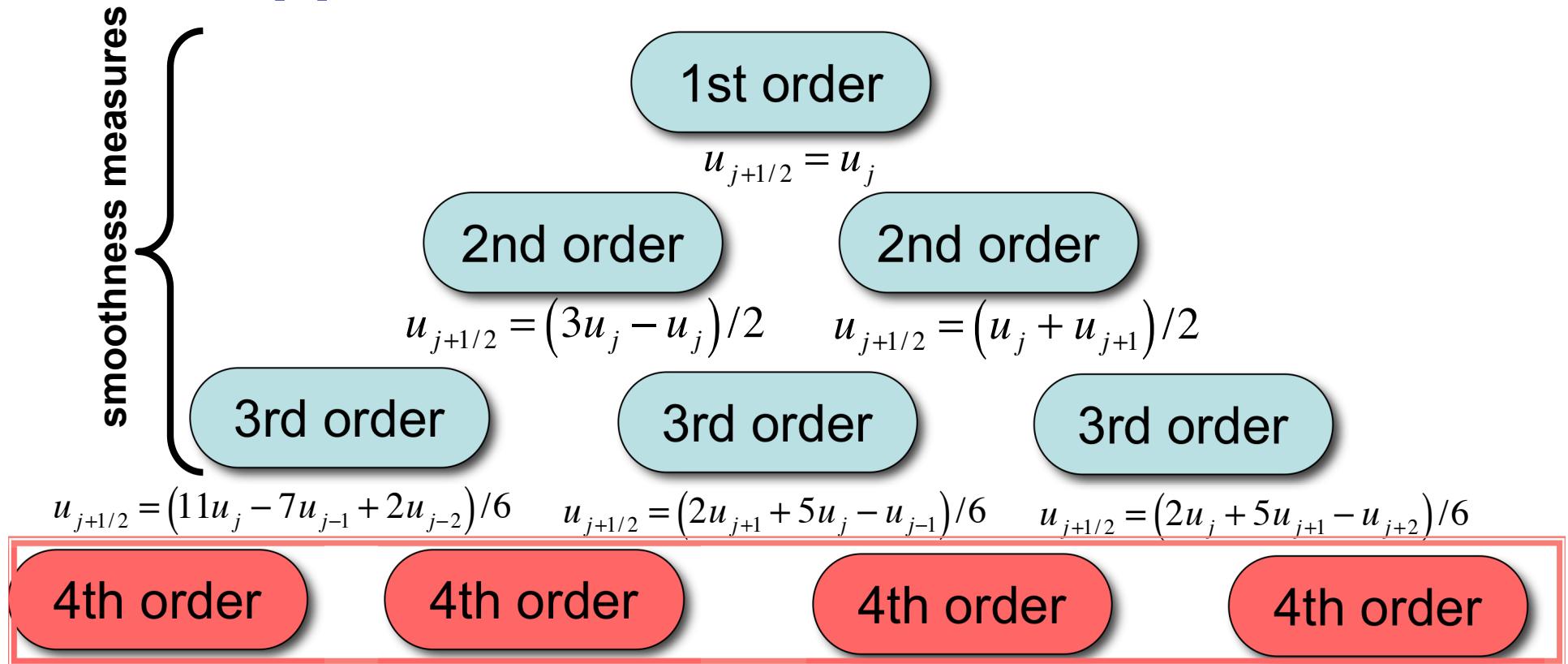
4th order

4th order

4th order

- The high-order stencils are evaluated pair-wise.

Weighted ENO methods are different in their approach, but the result is similar



- These methods evaluate ***all*** the high-order stencils and compare them algebraically.

Weighted ENO methods can have very high formal order of accuracy*.

- A nonlinear convex combination of schemes - three-3rd order to a 5th order, four-4th order to a 7th order...
- Nonlinearity comes in with smoothness detectors

$$f_{j+1/2} = \sum_m \omega_m f_{j+1/2,m}^m$$

- 3rd Order fluxes

$$f_{j+1/2,1} = \frac{1}{3}f_{j-2} - \frac{7}{6}f_{j-1} + \frac{11}{6}f_j$$

$$f_{j+1/2,2} = -\frac{1}{6}f_{j-1} + \frac{5}{6}f_j + \frac{1}{3}f_{j+1}$$

$$f_{j+1/2,3} = \frac{1}{3}f_j + \frac{5}{6}f_{j+1} - \frac{1}{6}f_{j+2}$$

- Constants to give 5th order

$$C_1 = 1, C_2 = 6, C_3 = 3$$

$$f_{j+1/2,\text{HO}} = \frac{1}{30}f_{j-2} - \frac{13}{60}f_{j-1} + \frac{47}{60}f_j + \frac{9}{20}f_{j+1} - \frac{1}{20}f_{j+2}$$

Weighted flux

$$\omega_k = \frac{w_k}{\sum_m w_m} \quad w_k = \frac{C_k}{(IS_k + \varepsilon)^p}$$

$$IS_k = \sum_{l=1}^{r-1} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} \left(q_k^{(l)} \right)^2 dx$$

$$IS_1 = \frac{13}{12} (f_{j-2} - 2f_{j-1} + f_j)^2 + \frac{1}{4} (f_{j-2} - 4f_{j-1} + 3f_j)^2$$

Weighted ENO methods can have very high formal order of accuracy*.

- A nonlinear convex combination of schemes - three-3rd order to a 5th order, four-4th order to a 7th order...
- Nonlinearity comes in with smoothness detectors

R-K Time Advance

$$u^{(1)} = u^n + \Delta t L(u^n) \quad f(u) = f^-(u) + f^+(u), f^\pm(u) = \frac{1}{2}(f(u) \pm \alpha u)$$

$$u^{(2)} = \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L(u^{(1)})$$

$$u^{n+1} = \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L(u^{(2)})$$

$$L = -\frac{1}{h}(f_{j+1/2} - f_{j-1/2})$$

Flux-Splitting

$$\omega_k = \frac{w_k}{\sum_m w_m} \quad w_k = \frac{C_k}{(IS_k + \varepsilon)^p}$$

$$f_{j+1/2} = \sum_m \omega_m f_{j+1/2,m}^m$$

Note: accuracy \neq efficiency!

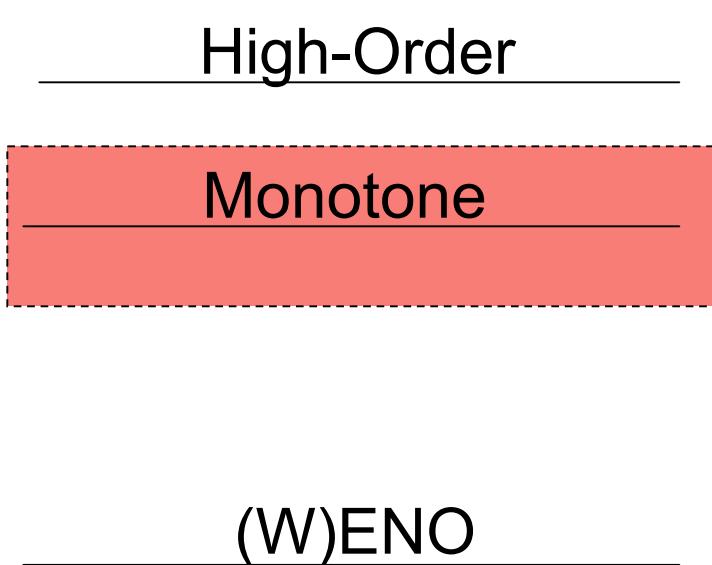
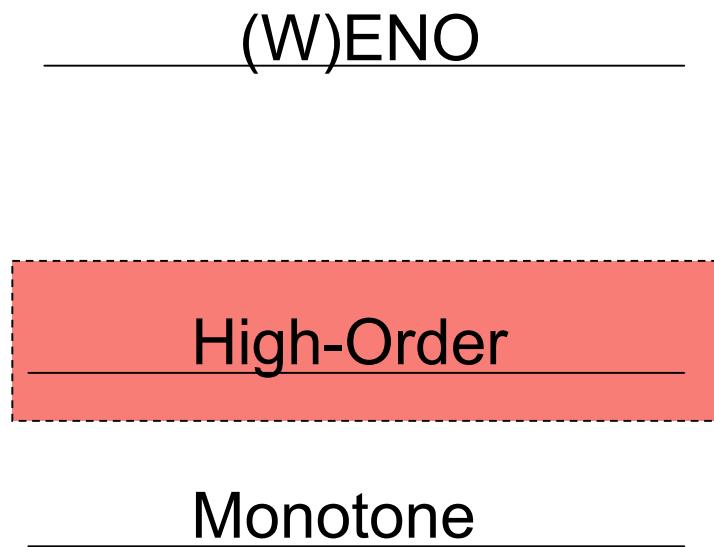
New algorithm development was motivated by the Greenough-Rider results.

- *Can we have the best of high-order, monotone- and ENO-type methods?*
- Hybridize the nonlinear monotone/non-oscillatory methods*
 - Start with a nonlinear monotone method: high-order + monotonicity test
 - If the flow is not monotone, then use the median of the
 - i) original high-order,
 - ii) monotone limiting value and
 - iii) an ENO/WENO value

*Similar to Huynh, *SIAM J. Num. Anal.*, 32 1995,
Suresh & Huynh, *J. Comp. Phys.*, 136 1997,
Daru & Tenaud, *J. Comp. Phys.*, 193, 2004

The new methods are based on bounding the approximation with nonlinearly stable methods using a median function.

Value of the approximation
(Edge value or slope)



The median() returns the value bounded by the other two values.

The median function has some key properties.

- One can use a median, or bounding function, **median**(a,b,c) that returns the middle argument of the three.
 - The one that is bounded by the other two
 - If two arguments are $O(h^n)$ the median is too!
 - If one argument is $O(h^n)$ and a second is $O(h^m)$ with $m < n$, the median is $O(h^m)$
 - If two arguments produce a linearly stable method, the **median** will as well.

The PPM method is based on polynomial interpolation.

- We find a local parabolic interpolant is
 $\mathbf{w}(x) = p(\theta) = p_0 + p_1\theta + p_2\theta^2; \quad \theta = (x - x_j)/\Delta x$
- where
$$p_0 = \frac{3}{2}\mathbf{w}_j - \frac{1}{4}(\mathbf{w}_{j-1/2} + \mathbf{w}_{j+1/2})$$
$$p_1 = \mathbf{w}_{j+1/2} - \mathbf{w}_{j-1/2}$$
$$p_2 = 3(\mathbf{w}_{j-1/2} + \mathbf{w}_{j+1/2}) - 6\mathbf{w}_j$$
- We describe a PPM that interpolates the characteristic variables rather than the primitive variables (ρ, u, p) of the original PPM.

Other high-order edge values can be used instead

First compute the edge values:

- Sixth-order centered stencil

$$\mathbf{w}_{j+1/2} = \frac{37(\mathbf{w}_j + \mathbf{w}_{j+1}) - 8(\mathbf{w}_{j-1} + \mathbf{w}_{j+2}) + (\mathbf{w}_{j-2} + \mathbf{w}_{j+3})}{60}$$

- Seventh-order upwind stencil

$$\mathbf{w}_{j+1/2} = \frac{-3\mathbf{w}_{j-3} + 25\mathbf{w}_{j-2} - 101\mathbf{w}_{j-1} + 319\mathbf{w}_j + 214\mathbf{w}_{j+1} - 38\mathbf{w}_{j+2} + 4\mathbf{w}_{j+3}}{420}$$

- Six-point optimal stencil $[0, 3\pi/4]$

$$\mathbf{w}_{j+1/2} = a(\mathbf{w}_j + \mathbf{w}_{j+1}) + b(\mathbf{w}_{j-1} + \mathbf{w}_{j+2}) + c(\mathbf{w}_{j-2} + \mathbf{w}_{j+3})$$

$$a=0.681056...; b=-0.229918..., c=0.048816..$$

- Produces optimally low phase error

ENO or WENO stencils could just as easily be used for the edge values.

- Stencils are precomputed (like WENO) and selected hierarchically using the differences in between stencils to select the smoothest (first 2nd order, then 3rd, then 4th, ...)

$$\mathbf{w}_{j+1/2}^{2nd} = \frac{(\mathbf{w}_j + \mathbf{w}_{j+1})}{2}, \frac{(^3\mathbf{w}_j - \mathbf{w}_{j-1})}{2}$$

$$\mathbf{w}_{j+1/2}^{3rd} = \frac{(^2\mathbf{w}_{j-2} - ^7\mathbf{w}_{j-1} + ^{11}\mathbf{w}_j)}{6}; \frac{(-\mathbf{w}_{j-1} + ^5\mathbf{w}_j + ^2\mathbf{w}_{j+1})}{6};$$

$$\frac{(^2\mathbf{w}_j + ^5\mathbf{w}_{j+1} - \mathbf{w}_{j+2})}{6}$$

In the original PPM, the edges are tested for their creation of a monotone interpolant.

- One follows these steps
 - Make sure that $w_{j+1/2}$ is between w_j and w_{j+1}
 - Next, make sure the polynomial is monotone, this amounts to making sure that $w_{j+1/2}$ is between w_j and $3w_j - 2w_{j-1}$
- Monotonicity can be implemented with two steps at each edge,

$$w_{j \pm 1/2} := \text{median}(w_j, w_{j \pm 1/2}, w_{j \pm 1})$$

$$w_{j \pm 1/2}^M = w_{j \pm 1/2} := \text{median}(w_j, w_{j \pm 1/2}, 3w_j - 2w_{j \mp 1/2})$$

One can replace this algorithm by a bounding estimate for monotonicity.

- This estimate does not use the high-order edge values, just the three closest cells – slightly less restrictive.

$$\mathbf{w}_{j\pm 1/2}^M = \mathbf{median}(\mathbf{w}_j, 3\mathbf{w}_j - 2\mathbf{w}_{j\mp 1}, \mathbf{w}_{j\pm 1})$$

$$\mathbf{w}_{j\pm 1/2} \doteq \mathbf{median}(\mathbf{w}_j, \mathbf{w}_{j\pm 1/2}, \mathbf{w}_{j\pm 1/2}^M)$$

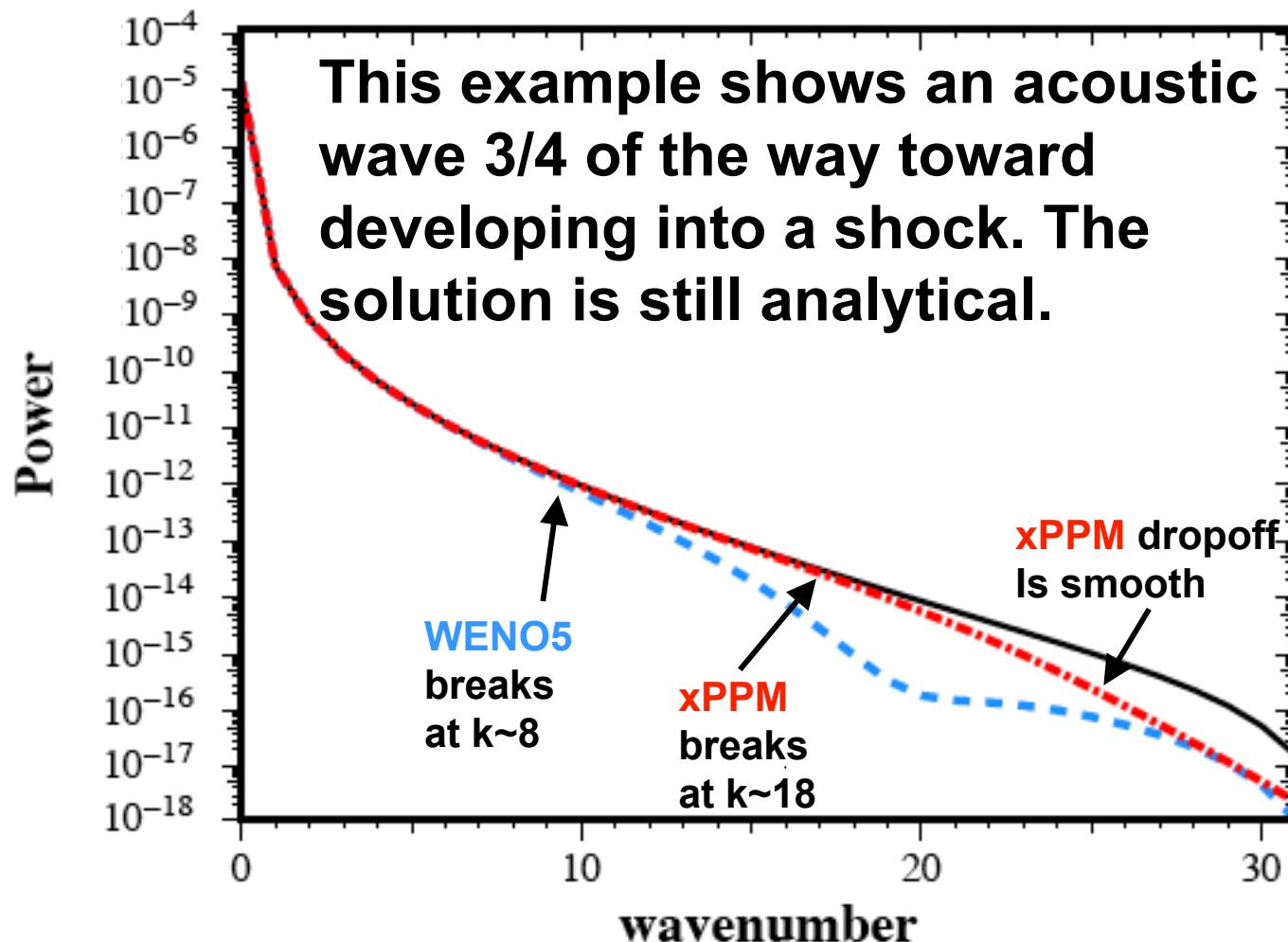
The monotonicity checking step is replaced for xPPM

- Check monotonicity
 - If the interpolant is monotone, return
 - If the interpolant is not monotone, continue
- Create a ENO or WENO approximation for the edge values
- The simplest version returns the median of the original high-order, W(ENO) and monotone limit.

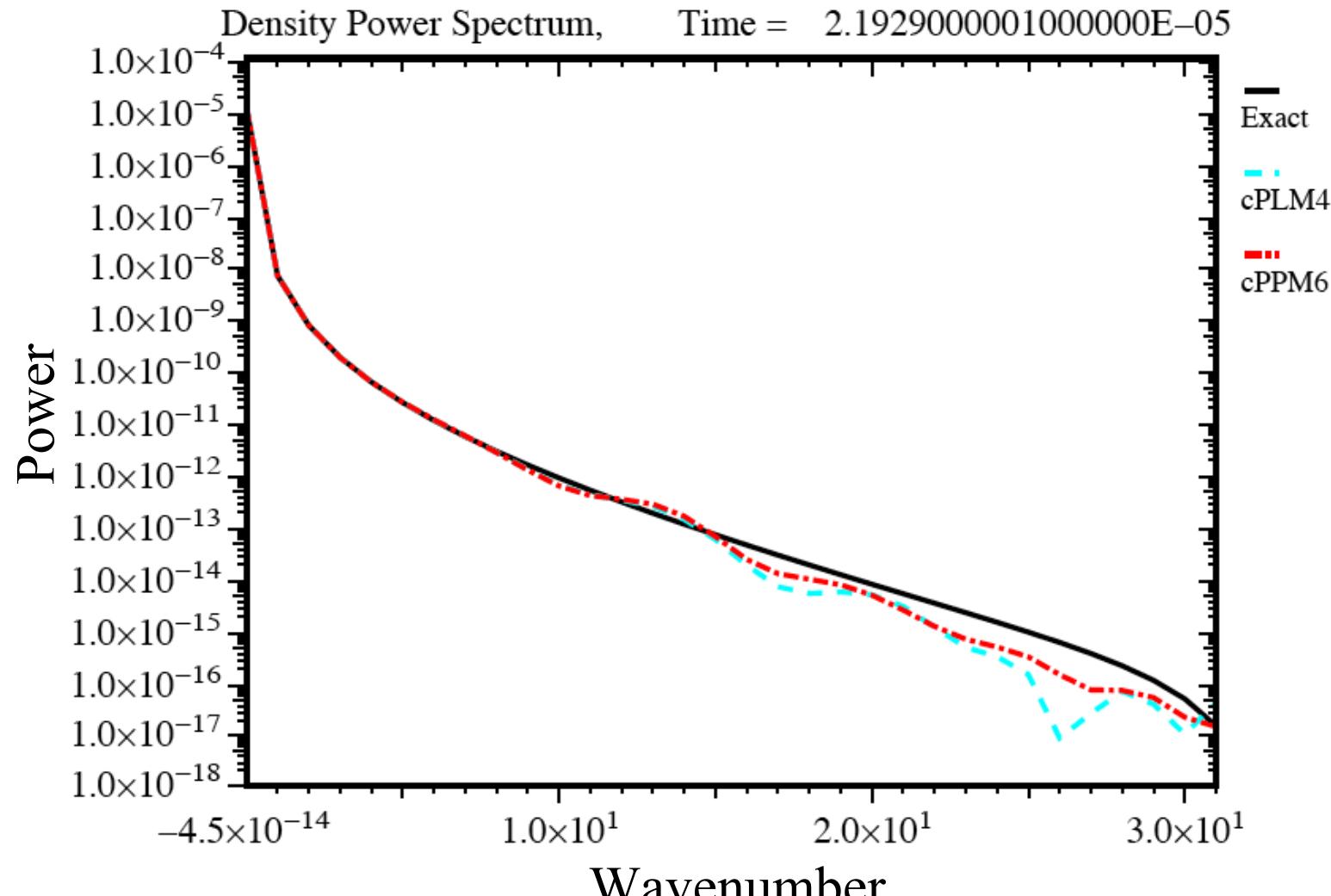
$$\mathbf{w}_{j\pm 1/2} = \mathbf{median}(\mathbf{w}_{j\pm 1/2}^{\text{H.O.}}, \mathbf{w}_{j\pm 1/2}^{\text{M}}, \mathbf{w}_{j\pm 1/2}^{\text{ENO or WENO}})$$

What's the impact? Look at a smooth wave-breaking problem spectrally.

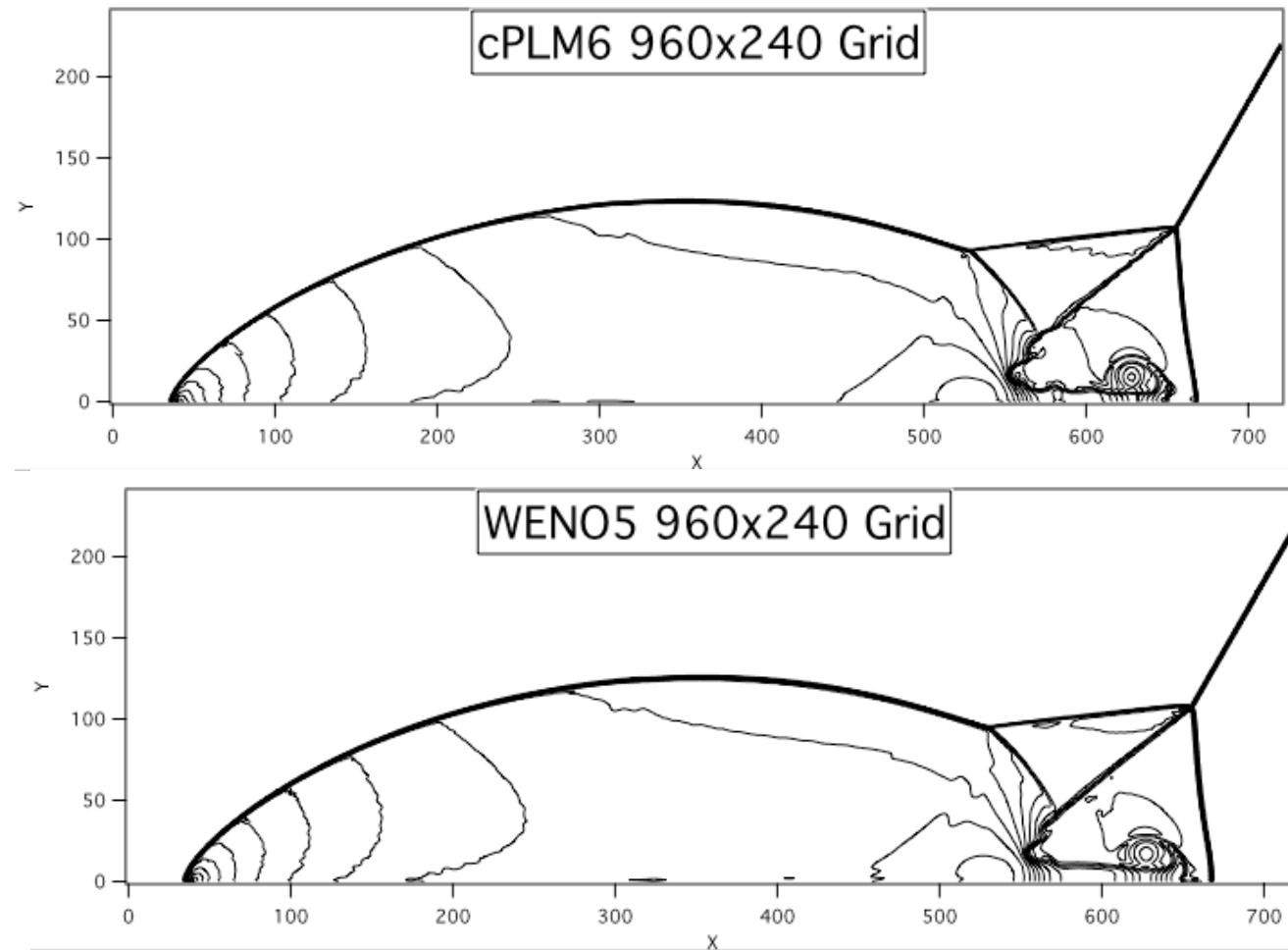
Compare the 5th order weighted essentially non-oscillatory method (WENO5) with our new extreme piecewise parabolic method (xPPM)



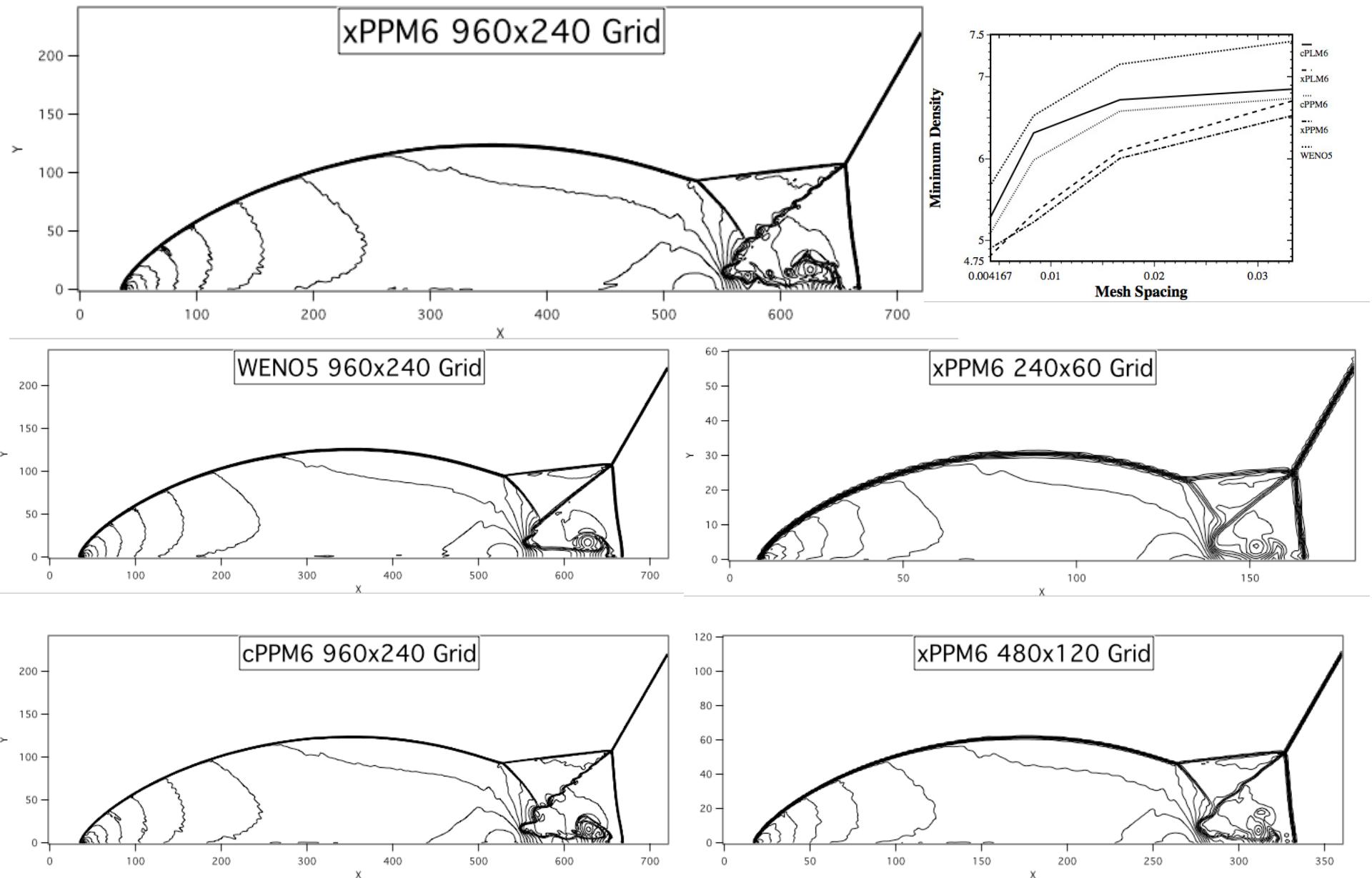
What's the impact? Look at a smooth wave-breaking problem spectrally



How do these methods do on computing the Mach 10 shock reflection problem*.



*A mach 10 shock reflecting from a 60 degree wedge.



Question: How does one get the most from a difference stencil?

- Use the best parts of different methods in combination where each performs “optimally.”
- Formal accuracy does not necessarily produce better or more efficient solutions; however, high-order algorithmic elements do substantially improve algorithmic efficiency.
- Form approximations that are accurate and nonlinearly stable using bounding principles.
- *Computational efficiency is always a consideration to keep in mind and measure.*